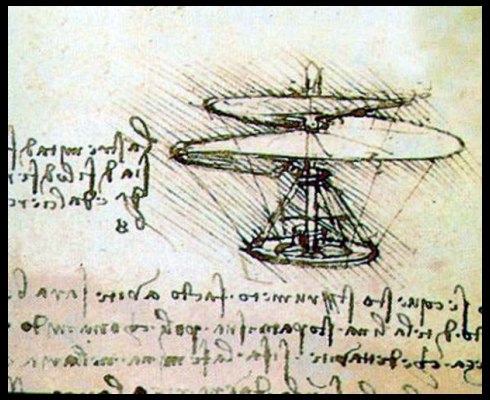

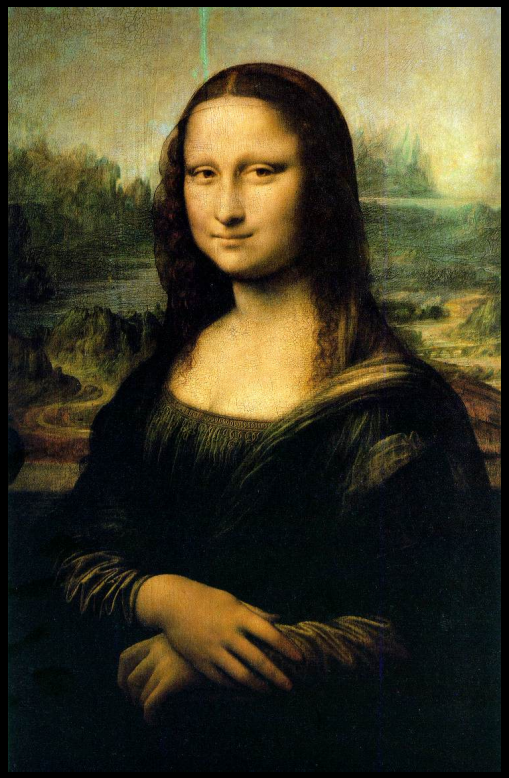

*Development tools for **art***

Chris Green
SCD-ADSS-SSI

Fermilab
Fermi National Accelerator Laboratory
Office of Science / U.S. Department of Energy
Managed by Fermi Research Alliance, LLC

Outline

- Motivation.
- Feature overview.
- Details.
- Developer experience.
- Ongoing developments.

Why more build tools?

- Package / install.
  Intensity Frontier experiments want the switchability of UPS but without the `ups declare` commands, and the ability to install multiple packages from tarball.
- Build.
  - Parallel builds.
  - Support for testing: scripts, execs, packages, dependent tests, pass / fail criteria.
  - Build consistency (very important).
  - Ease of setup / use.
  - Must be able to provide packages usable by other build system (vital).

Feature overview

- Use existing products as basis: **UPS**, **CMake**, **CTest**, **GNU Make**.
- Hierarchical package system with rigid version and qualifier dependency checking to ensure binary-compatible packages and consistent applications (compiler version, debug, language standard).
- Easy install of all or part of linked packages via tarball.
- Simple developer setup, tailored directives to specify build products, dependencies, tests, *etc*..
- Seamless parallel build / test.
- Non-viral: **art** suite is **UPS** packages, use of **CMake** not required by experiments.
- All builds out-of-source: same source supports co-existing debug / profile builds.

Details: ***Relocatable UPS***

- Same **UPS**, new capabilities.
- No `prd/`, `db/` directories
- Each product version has a `${PROD_VER}.version` directory instead.
- Use product dependencies (with `-B` option, `+` qualifiers) to ensure compatible products with error-on-failure.

Details: ***CMake***

- Built-in support for parallel builds, test.
- Flexible, modular, higher-level than **GNU Make**.
- Excellent dependency management.
- Install, package, package configuration facilities mesh well with **Relocatable UPS**.

Details: ***cetbuildtools***

- **CMake**-based.
- Simple product configuration with `product_deps` file, listing external package dependencies.
- Straightforward **CMake** macros to do most common things: libraries, execs, **art** plugins, **ROOT** dictionaries, tests with pass / fail criteria *etc.*.
- Each package built separately and installed as a **Relocatable UPS** product to guarantee a consistent build.
- Auxiliary scripts for things like version bumps, coherent release builds, *etc.*.

Developer experience

- Used to build, package and deliver the **art** suite to **Intensity Frontier** experiments.
- Being used on a small-scale by **Muon g-2**.
- Full **art** suite build / test / install / package with > 200 tests in 8:20 (vs 48:55 serial). A NOP build of **art** only followed by tests is 0:34 (4:04 serial).
- Able to specify unit tests up through integration tests very easily, including arguments, dependent test chains, required files, *etc.*.

Ongoing development

- Automatic generation of a package's installed library list for use by other packages.
- A safe scheme for reliable simultaneous multi-package builds: test releases *a la* **SoftRelTools** are notorious for allowing inconsistent builds. This in turn may necessitate:
- Library versioning / checking scheme integral to the build system.
- Sell to **Intensity Frontier** experiments.